TECHNICAL MEMORANDUM 1891

RELY, A PROGRAM FOR ESTIMATING OVERALL SYSTEM RELIABILITY BASED ON COMPONENT, SYSTEM AND FLIGHT DATA

THOMAS A. NEEF

MARCH 1969

PICATINNY ARSENAL
DOVER, NEW JERSEY

TECHNICAL MEMORANDUM 1891

RELY,

A PROGRAM FOR ESTIMATING OVERALL SYSTEM RELIABILITY BASED ON COMPONENT, SYSTEM AND FLIGHT DATA

BY

THOMAS A. NEEF

MARCH 1969

DATA PROCESSING SYSTEMS OFFICE

PICATINNY ARSENAL

DOVER, NEW JERSEY

TABLE OF CONTENTS

ABSTRACT

This report documents a program for estimating overall system reliability by judiciously combining individual component data, laboratory test data and finally flight data as gathered at two different periods in time. All data used in arriving at this single reliability estimate is assumed to be attribute in nature.

INTRODUCTION

In July 1968, the Munitions Command requested the Mathematical Analysis Division of the Data Processing Systems Office to develop a computer program (using their prescribed methodology) to estimate overall system reliability. Such a program has been created, herein called RELY, which requires as input six sets of distinct attribute type data which should commonly arise during the development phase of a missile assembly or subassembly program. The six sets of basic inputs consist of the three types of data listed below gathered at two distinct time periods:

(1) Component data as naturally arising from individual component tests.

(2) Laboratory systems data arising from testing a component assembly (i.e. circuit) in a laboratory and

(3) Flight data consisting of data resulting from actual test firings containing all circuitry.

The program judiciously combines this data so as to produce a single estimate for overall system reliability.

This report describes in detail this computer program and includes a complete description of input-output formats, a discussion of the program logic and a sample case. It should be emphasized however, that this report does not describe the foundations of the methodology utilized. For this the interested reader is referred to Reference 1.

GENERAL PROGRAM OUTLINE

To simplify the discussion the following mathematical notation has been adopted.

Mathematical Notation:

A_{ij}	A = N refers to number of tests conducted at i th stage at time T_j
(basic inputs)	A = C refers to number of successes resulting from testing at the i th stage at time T_j

Subscript definition:

i = c (Component stage) refers to component data

i = s (System stage) refers to system data, i.e. resulting from testing circuits comprised of the above components

i = f (flight stage) refers to flight data, i.e. resulting from flight tests containing the circuitry above

j = o refers to time T_o

j = 1 refers to time T_1 where $T_o < T_1$

Thus N_{s1} refers to the number of systems tests conducted at time T_1, while C_{fo} refers to the number of flight successes obtained at time T_o.

K_i	refers to a weighting factor that weights the less significant T_o data when combining with T_1 data. i again refers to component (i=c), systems (i=s) or flight (i=f) data.

A_{ieq} refers to the equivalent number of tests (A=N:, or equivalent number of successes (A=C) for either component, system or flight data, (i=c, s, f respectively). These values arise by appropriate combination of N_{co}, C_{co}, N_{c_1}, C_{c_1}, K_c to obtain N_{ceq} and C_{ceq}, or more generally by combination of A_{io}, A_{i_1} and K_i, to produce A_{ieq}.

K_{ieq} refers to the weight given to the equivalent component data, when combining with the equivalent system data (i=s), to produce combined component-system data. When i=f, this variable refers to the combination of the component-system data with the equivalent flight data to produce the overall equivalent number of tests N_{eq} and overall equivalent number of successes C_{eq} used to arrive at P_{best}.

P_{best} refers to the overall system reliability = C_{eq}/N_{eq}

The interrelationships existing between the various data sets are exhibited in Figure 1. Figure 2 is a schematic diagram showing the relative significance of the involved data sets.

The underlying approach to arrive at P_{best}, the single estimate

of overall reliability, is to combine all like data sets, degrading however, the less significant T_o data, when combining with the more significant T_1 data. Similar degradation factors apply when combining the less significant equivalent component data, with the more significant equivalent system data and the more significant, yet, flight data.

Each such combination of data will be described, thus yielding in the process a general program outline.

I. Combination of component data N_{co}, C_{co}, N_{c_1}, C_{c_1} to produce N_{ceq}, C_{ceq}.

A. One component case.

Here N_{co}, C_{co}, N_{c_1}, C_{c_1} embody the inputs to this calculation. The values for the above are obtained from actual testing, witnessing for example C_{co} success out of N_{co} tests at time T_o.

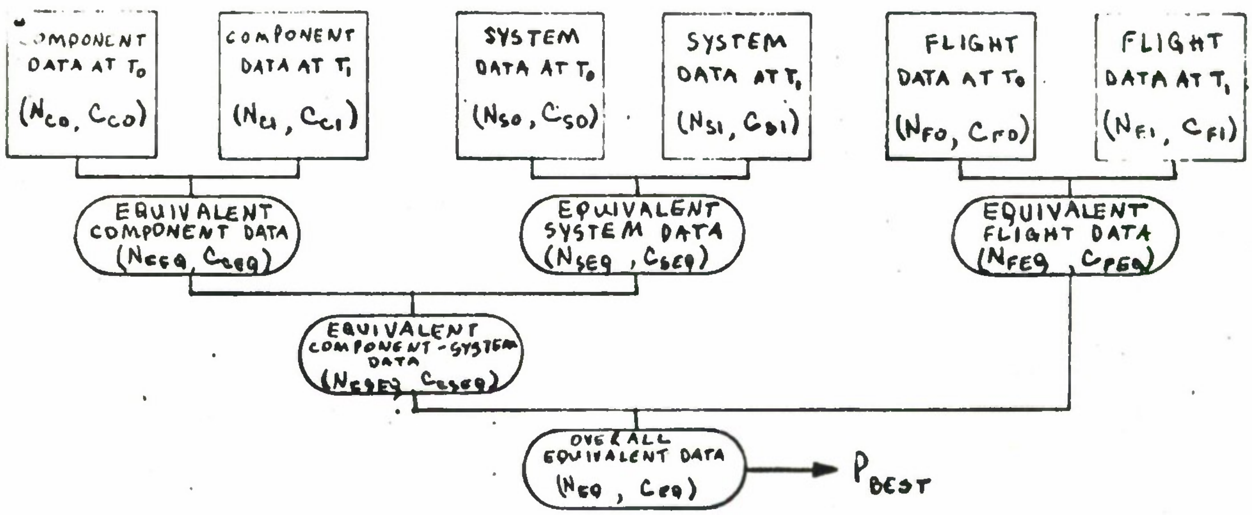
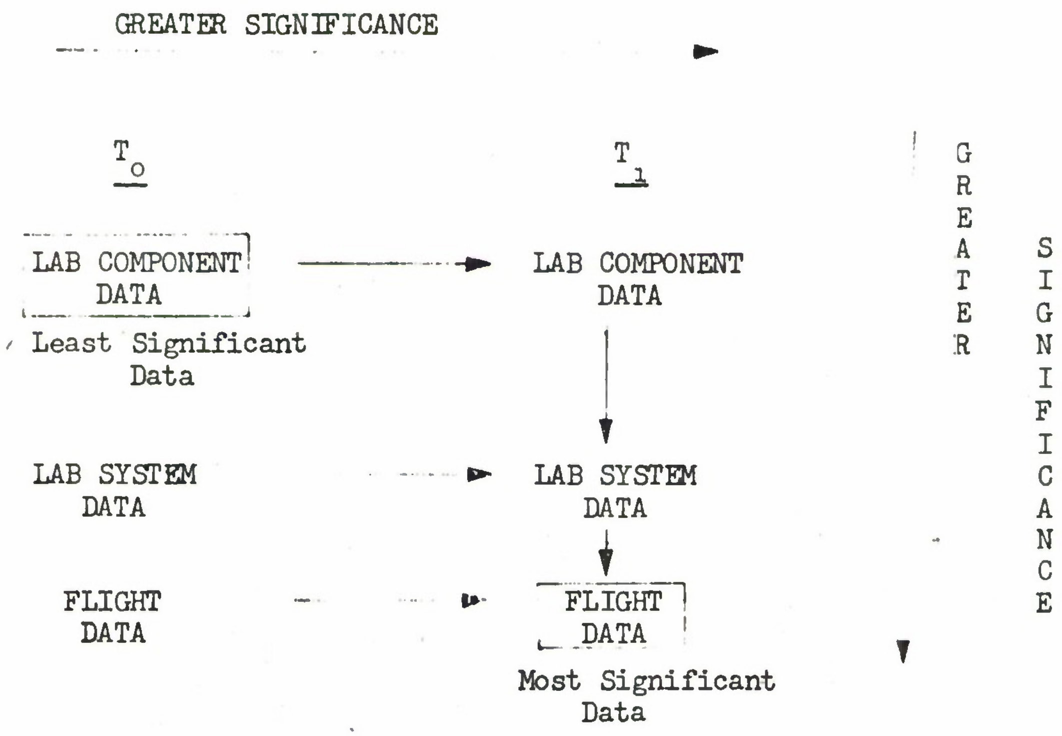


FIGURE 1



RELATIVE SIGNIFICANCE OF DATA

FIGURE 2

It is assumed that P, the true probability of component functioning is β-distributed with N_{co}, C_{co} as parameters, i.e.

$$\beta_o(P) = \frac{\Gamma(N_{co}+2)}{\Gamma(C_{co}+1)\ \Gamma(N_{co}-C_{co}+1)} P^{C_{co}}(1-P)^{N_{co}-C_{co}} \tag{1}$$

with a similar expression $\beta_1(P)$ for components tested at time T_1. The T_o data is degraded by the area, K_c, common to the two β-distributions, as expressed in the following equations that yield the equivalent component data.

$$N_{ceq} = N_{co} \cdot K_c + N_{c1}; \quad C_{ceq} = C_{co} \cdot K_c + C_{c1} \tag{2}$$

K_c is interpreted geometrically in the Diagram below

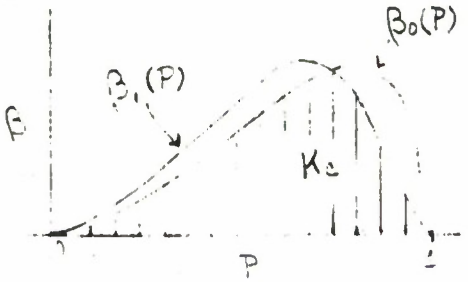


DIAGRAM 1

Thus if $N_{co} = N_{c1}$ and $C_{co} = C_{c1}$ the two distributions would coincide and the degradation factor would be 1 (i.e. no degradation).

B. Several Component case.

When several different component types are involved and related by a specific circuit equation, calculations additional to those above are necessary. In this case it is required to compute a N_{ceq} and C_{ceq} for each component type and combine the results. Let $N_{ceq}(k)$ and

$C_{ceq}(k)$ represent the above values referred to the k-th component type. An equation similar to equation (1) is constructed with these values i.e.

$$\beta(P,k) = \frac{\Gamma(N_{ceq}(k) + 2)}{\Gamma(C_{ceq}(k) + 1)\,\Gamma(N_{ceq}(k) - C_{ceq}(k) + 1)}\,Q \tag{3}$$

where

$$Q = P^{C_{ceq}(k)}\,(1 - P)^{N_{ceq}(k) - C_{ceq}(k)}$$

This represents the probability density P of the k-th component type in the light of both T_o and T_1 data. The calculation to combine each $N_{ceq}(k)$ and $C_{ceq}(k)$ to produce N_{ceq} and C_{ceq} then proceeds in a Monte Carlo fashion involving the circuit.

Values are sampled from each of the β-distributions* and inserted into the specified circuit equation, from which the probability of circuit functioning, R_s for these values, is obtained. It is important to note that if a given component type appears several times in the circuit, the corresponding β-distribution is sampled only once per simulation. Performing this simulation M times thus yields M values for the probability of circuit functioning. The assumption is made that these M values are β-distributed. Consequently the mean μ and the variance σ^2 of these numbers are computed and set equal to the mean and variance of the β-distribution.

*This is accomplished by numerically determining the cumulative distribution and finding the P corresponding to RN = $\int_0^1 \beta(k,X)\,dX$ where RN is a random number between 0 and 1. The integral is tabulated as a function of P, so that for given RN, linear interpolation applies for ascertaining P.

This allows for the two unknown parameters in this distribution to be solved. These unknowns correspond in fact to the N_{ceq} and C_{ceq} of Figure 1. Mathematically, this procedure is equivalent to the equations that follow.

$$\mu = \frac{\sum_{i=1}^{M} R_s(i)}{M} \qquad \sigma^2 = \frac{\sum_{i=1}^{M} R_s^2(i) - 2\mu \sum_{i=1}^{M} R_s(i) + M\mu^2}{M} \tag{4}$$

$$N_{ceq} = \frac{\mu(1-\mu)}{\sigma^2} - 3 \qquad C_{ceq} = \frac{\mu^2(1-\mu)}{\sigma^2} - (1+\mu) \tag{5}$$

II. Combination of system data N_{so}, C_{so}, N_{s1}, C_{s1} to produce N_{seq} and C_{seq},

This combination is accomplished analogously to equations (1) where the degradation factor is again the area common to the two distributions. Thus

$$\beta_0(P) = \frac{\Gamma(N_{so}+2)}{\Gamma(C_{so}+1)\,\Gamma(N_{so}-C_{so}+1)} P^{C_{so}}(1-P)^{N_{so}-C_{so}}$$

$$\beta_1(P) = \frac{\Gamma(N_{s1}+2)}{\Gamma(C_{s1}+1)\,\Gamma(N_{s1}-C_{s1}+1)} P^{C_{s1}}(1-P)^{N_{s1}-C_{s1}}$$

$$K_s = \int_0^1 \mathrm{Min}\{\beta_0(P), \beta_1(P)\}\, dP \tag{6}$$

$$N_{seq} = N_{so} \cdot K_s + N_{s1}$$

$$C_{seq} = C_{so} \cdot K_s + C_{s1}$$

III. Combination of flight data N_{fo}, C_{fo}, N_{f1}, C_{f1} to produce N_{feq}, C_{feq}.

This again is completely analogous to II with N_{so}, ..., C_{s1} replaced by N_{fo}, ..., C_{f1} respectively.

IV. Combination of equivalent data produced in I, II, and III to produce N_{eq}, C_{eq}.

The combination of N_{ceq}, C_{ceq} with N_{seq}, C_{seq} to produce $N_{c,seq}$, $C_{c,seq}$ and the combination of these last two values with N_{feq}, C_{feq} to produce N_{eq}, C_{eq} are also completely analogous to II with the A_{ceq} replacing the A_{so} and the A_{seq} replacing the A_{s1} in the first case and the $A_{c,seq}$ replacing the A_{so} and the A_{feq} replacing the A_{s1} in the second case. Thus the last equation in (6) becomes $C_{c,seq} = C_{ceq} \cdot K_{seq} + C_{seq}$ when combining components and system data, and becomes $C_{eq} = C_{c,seq} \cdot K_{feq} + C_{feq}$ when combining component-systems data with flight data.

V. Calculation of overall system reliability, P_{best}.

$$P_{best} = \frac{C_{eq}}{N_{eq}} \tag{7}$$

For output purposes, the following two variable are also computed by the equations that follow

$$Mu = \frac{C_{eq} + 1}{N_{eq} + 2} \qquad VARIANCE = \frac{(C_{eq} + 1)(N_{eq} - C_{eq} + 1)}{(N_{eq} + 2)^2 (N_{eq} + 3)} \tag{8}$$

COMPUTER PROGRAM DESCRIPTION

A. Description of MAIN program

This Section describes the computer program that evaluates the mathematical model. It is believed that sufficient information is contained herein along with COMMENT statements in the program listing to enable one to understand and possibly modify the program. Figure 3 is an overall flow chart giving the sequence of the computations. Not shown in the figure is the fact that much of the routine calculations such as the tabulation of the β- distribution, numerical integration etc. are relegated to subroutines. These subroutines though briefly referred to in the text are more fully described at the end of this Section.

All inputs are read in the MAIN program, thereupon one set of component data is selected, being the values for N and C at time T_o and values for N and C at T_1. Using these four values (labelled in the program as NCO, CCO, NC1, CC1 respectively) as input arguements the value KA (=K_c of Diagram 1) is ascertained from a subprogram K. The T_o data is then degraded by this factor and the result added to the T_1 data to yield two new intermediate values, SUB1, corresponding to the number of trials and SUB2 corresponding to the number of successes as in equation (2) (page 7). Using these intermediate values as parameters a β- distribution (corresponding to equation 3) is constructed and tabulated in subroutine BETAF. The tabulation is effected at equally spaced abscissas depending upon the value of DELTAP which should lie

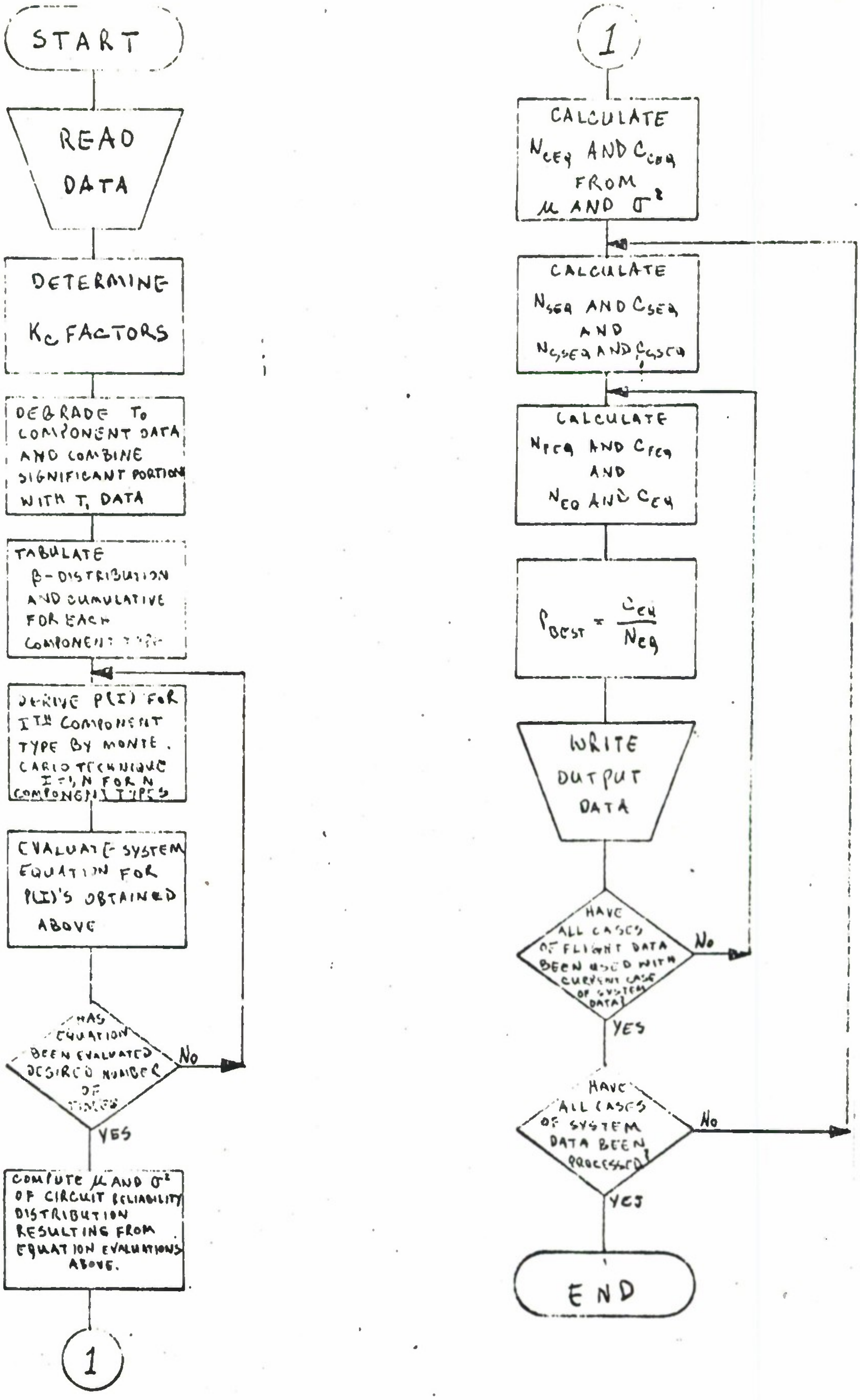

START
READ DATA
DETERMINE K_c FACTORS
DEGRADE TO COMPONENT DATA AND COMBINE SIGNIFICANT PORTIONS WITH T_1 DATA
TABULATE β-DISTRIBUTION AND CUMULATIVE FOR EACH COMPONENT
DERIVE P(I) FOR I^{th} COMPONENT TYPE BY MONTE CARLO TECHNIQUE I=1,N FOR N COMPONENT TYPES
EVALUATE SYSTEM EQUATION FOR P(I)'S OBTAINED ABOVE
HAS EQUATION BEEN EVALUATED DESIRED NUMBER OF TIMES
NO
YES
COMPUTE μ AND σ^2 OF CIRCUIT RELIABILITY DISTRIBUTION RESULTING FROM EQUATION EVALUATIONS ABOVE.
1
1
CALCULATE N_{CEQ} AND C_{CEQ} FROM μ AND σ^2
CALCULATE N_{SEQ} AND C_{SEQ} AND N_{GSEQ} AND C_{GSEQ}
CALCULATE N_{FEQ} AND C_{FEQ} AND N_{EQ} AND C_{EQ}
$P_{BEST} = \frac{C_{EQ}}{N_{EQ}}$
WRITE OUTPUT DATA
HAVE ALL CASES OF FLIGHT DATA BEEN USED WITH CURRENT CASE OF SYSTEM DATA?
NO
YES
HAVE ALL CASES OF SYSTEM DATA BEEN PROCESSED?
NO
YES
END

Figure 3

in the interval [.01, .5]. It is from this density function that the probability of component functioning is sampled and subsequently inserted into the circuit equation. To accomplish this, the cumulative of this distribution is tabulated and placed in an array A(I) (subroutine DQSF), a random number, (RDM(1)), selected and the corresponding P value determined by linear interpolation in subprogram INTERP. This calculation is repeated for each component type, and all results are ultimately substituted into the circuit equation from which a probability of circuit functioning, (RS), is computed. At this point a running count of these RS values for output purposes in histogram form is maintained, along with running summations of the RS values and RS^2 values for use in calculating the mean μ and variance σ^2 of these generated numbers (equations 4). At the end of the simulation loop, equations (5) are used to produce N_{ceq} and C_{ceq} which are labelled as NCEQ and CCEQ in the program. Remaining calculations involving system and flight data, and all combinations thereof are analogous to what has been described and again typically involve use of the BETAF and K routines. The notation, too, in remaining segments of the MAIN program is quite suggestive so that the reader will find for example N_{so}, C_{seq}, and P_{best} are represented in the program as NSO, CSEQ and PBEST.

B. Description of Subprograms (in alphabetical order)

1. FUNCTION AREAF (B,J)

Common DELTAP

Description of arguments:

B - one-dimensional equidistantly tabulated function the area under which is to be computed.

J - the number of values in B.

DELTAP - interval width

Operation:

Calculates the area under B by Simpson's rule, (therefore J must be odd). The output is a scalar representing the area under the curve.

2. SUBROUTINE BETAF (N, C, BETA, J, PB)

Common DELTAP

Description of arguments:

Input Arguments:

N,C - input parameters for the beta-distribution equation

Output Arguments:

BETA - one dimensional beta-distribution array produced in the subroutine

J - number of values in array (dependent upon DELTAP)

PB - one dimensional array of P-values used in generating the beta distribution

DELTAP - interval width

Operation:

Produces an equidistantly tabulated beta-distribution

using the following equation:

$$\beta = G\, P^{C} \;(1 - P)^{N - C} \tag{9}$$

where G is a constant,

$$G = \frac{\Gamma(N + 2)}{\Gamma(C + 1)\;\Gamma(N - C + 1)} = \frac{(N+1)!}{C!\,(N-C)!} \tag{10}$$

and is computed in subroutine GAMA, P varies from 0.0 to 1.0 in increments of DELTAP

In order to avoid problems arising from zero bases and/or zero exponents in the exponentiation routine, the equation is evaluated for one of four cases as:

1. N and C are distinct and nonzero.
2. N and C are identical and nonzero.
3. N is nonzero but C is zero.
4. Both N and C are zero.

3. SUBROUTINE DQSF (H, Y, Z, NDIM)

This subprogram is part of the IBM System/360 SCIENTIFIC SUBROUTINE PACKAGE and a detailed description of its arguments and operation can be found therein. Briefly, however, this subroutine performs the integration of an equidistantly tabulated function by Simpson's Rule. It computes a vector of integral values Z_i for a table of functional values Y_i, i= 1, 2, ..., N given at equidistant points $X_i = a + (i-1)h$

$$Z_i = Z(X_i) = \int_a^{X_i} Y(X)\,dx \qquad (i= 1, 2, 3, \ldots, N)$$

4. FUNCTION GAMA (ARG1, ARG2, ARG3)

Description of arguments:

ARG1 - corresponds to N+2 in equation (10) of subroutine BETAF

ARG2 - corresponds to C+1 in equation (10) of subroutine BETAF

ARG3 - corresponds to N-C+1 in equation (10) of subroutine BETAF

Operation:

The IBM - supplied subroutine GAMMA evaluates the GAMMA function for a given argument provided the argument is less than or equal to 57. The subroutine GAMA is designed to reduce each of the arguments ARG1, ARG2, ARG3, to 57, if necessary, before calling GAMMA for each of them. Furthermore it computes G

$$G = \frac{\Gamma(ARG1)}{\Gamma(ARG2)\ \Gamma(ARG3)} \tag{11}$$

for use in subroutine BETAF. Output is scalar value for G.

5. SUBROUTINE GAMMA (XX, GX, IER)

This subroutine is part of the SCIENTIFIC SUBROUTINE PACKAGE except that was changed to double precision for this program. Briefly the operation of this subroutine is to evaluate the gamma function for a given value of XX, where Γ (X) is defined for X > 0 by

$$\Gamma(X) = \int_0^{\infty} t^{X-1} e^{-t}\, dt$$

which satisfies the recurrence relation $\Gamma(X) = (X-1)\Gamma(X-1)$

which defines $\Gamma(X)$ for any X non-negative integer i.e.

$\Gamma(X) = (X-1)!$

6. FUNCTION INTERP (I, J, F, YFL)

Common /BLK1/AX

Description of arguments:

I - component type designation

J - number of values in ITH row of two-dimensional array AX corresponding to array AREA of the MAIN program, in which the scalar input YFL will be interpolated.

F - the one-dimensional array from whose elements the interpolated value is to be calculated.

YFL - number whose corresponding value is to be found by interpolation

AX - subroutine name for array AREA of MAIN program which stores cummulative for each component type.

Operation:

Straight-forward linear interpolation is effected herein.

7. FUNCTION K (NO, CO, N1, C1)

Common DELTAP

Description of arguments

NO, CO - number of trials and number of successes at time T_o (or analogous parameters)

N1, C1 - number of trials and number of successes at most recent testing T_1 (or analogous parameters)

DELTAP - interval width

Operation:

K is designed to produce a proportionality facotr for degrading the less significant data, NO, CO (before it is combined with the more significant data N1, C1 in MAIN program). This is accomplished by first tabulating the beta-distributions for NO, CO and for N1, C1 by calling subroutine BETAF. The area under the intersection of these two curves, CMAREA, is found next by comparing the two beta curves point-for-point, in each case taking the smaller of the two values (ordinates), putting it into a new array, and computing the area under this curve in subroutine AREAF. AREAF is also used to find the area, TIAREA, under the beta curve for the most significant data N1, C1 (this area should be close to or exactly equal to 1.0 and is computed and used as a self adjusting calculation to account for digital round-off error). K is then set equal to CMAREA divided by TIAREA. The output of this subroutine is the scalar significance factor, K.

PROGRAM OPERATION

This Section describes the inputs necessary to run the program RELY as well as a description of the output variables. To allow for the desireability of analyzing several different system and/or flight data sets for a given set of component data, the inputs have been so arranged to facilitate these additions with a minimum of card entries. One may note, further, that since the bulk of the calculations are concerned with component data, additional results depending on varying system or flight data sets require only a small investment of computer running time for the added evaluation.

To operate the program RELY the probability equation for the circuit must be inserted into the program between the COMMENT statements MATH MODEL-BEGIN and MATH MODEL-END (see program listing). In doing this one should number each component type in any convenient manner so that a correspondence exists between the component test results and the same component in the circuit. An example for a simple circuit is contained in the sample inputs. In addition values for the variables listed below must be supplied in the format indicated, as well as the actual component, system and flight input data.

VARIABLE	DESCRIPTION	FORMAT
T	Time between testing dates (months).	I2
NRS	Number of times the circuit equation is to be evaluated (NRS>1) i.e. number of Monte Carlo simulations, limited only by format size and computer running time.	I6

VARIABLE	DESCRIPTION	FORMAT
DELTAP	(Δ P) Desired increment on P-values in setting up Beta-distributions and reliability distribution ($.01 \leq \Delta P \leq .5$)	F6.4
NCOMP	Number of distinct component types in the circuit ($1 \leq$ NCOMP ≤ 100)	I4
NSYDT	Number of system data sets to be used with a given set of component data ($1 \leq$ NSYDT ≤ 100)	I4
NFLDT	Number of flight data sets to be used in conjunction with the given component data set ($1 \leq$ NFLDT ≤ 100)	I4

Values for these six variables form the first card of the data deck. This card is followed by: the component data set, which contains one card for each component type, the laboratory system data set and finally the flight data set. Each card of each set has the same format (4I6). The four values on these cards are N_{io}, C_{io}, N_{i1}, C_{i1} in that order. The first card of the component data set must contain the data corresponding to the first component as labelled in the circuit equation, the second card contains data for component 2, and so forth for all component data.

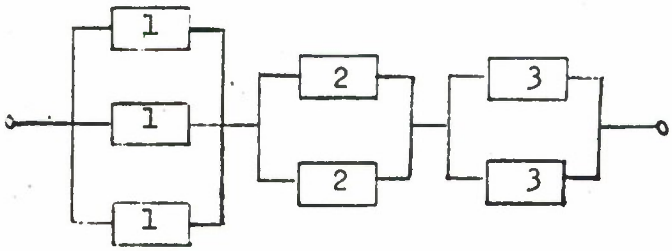
All data cards of the system and flight data sets are set up exactly like those of the component data set, the order of cards within each set being entirely arbitrary. All combinations of system and flight data, will be evaluated when NSYDT>1 and NFLDT>1, all cases of flight data being combined in turn with each case of system data. The program may be run without system and/or flight data. In this case NSYDT and/or NFLDT must be set equal to 1, and the appropriate data set must

consist of one blank card.

For a description of the output of program RELY, refer to the sample case, page 25.

SAMPLE CASE

As a sample case consider the simple circuit configuration below. It consists of seven components but only three distinct component types: type 1, type 2, type 3.



The probability equation for this circuit is

$$R_s = (1 - (1 - P(1))^3) (1 - (1 - P(2))^2) (1 - (1 - P(3))^2)$$

where P(i) is the probability of component type i functioning properly.

Typical input data for this circuit is presented in the table on the next page. It will be noted, for example, that components of type 2 worked satisfactorily 42 times out of 44 trials during the first round of testing (T_0) but twenty four months later (T_1), worked only 38 times in 44 trials.

In order to compute an estimate of overall system reliability for this circuit based on the data compiled in the following table using program RELY it is necessary to:

1. Insert the circuit equation into the program between the comment cards MATH MODEL - BEGIN and MATH MODEL - END (See Listing which contains this example).

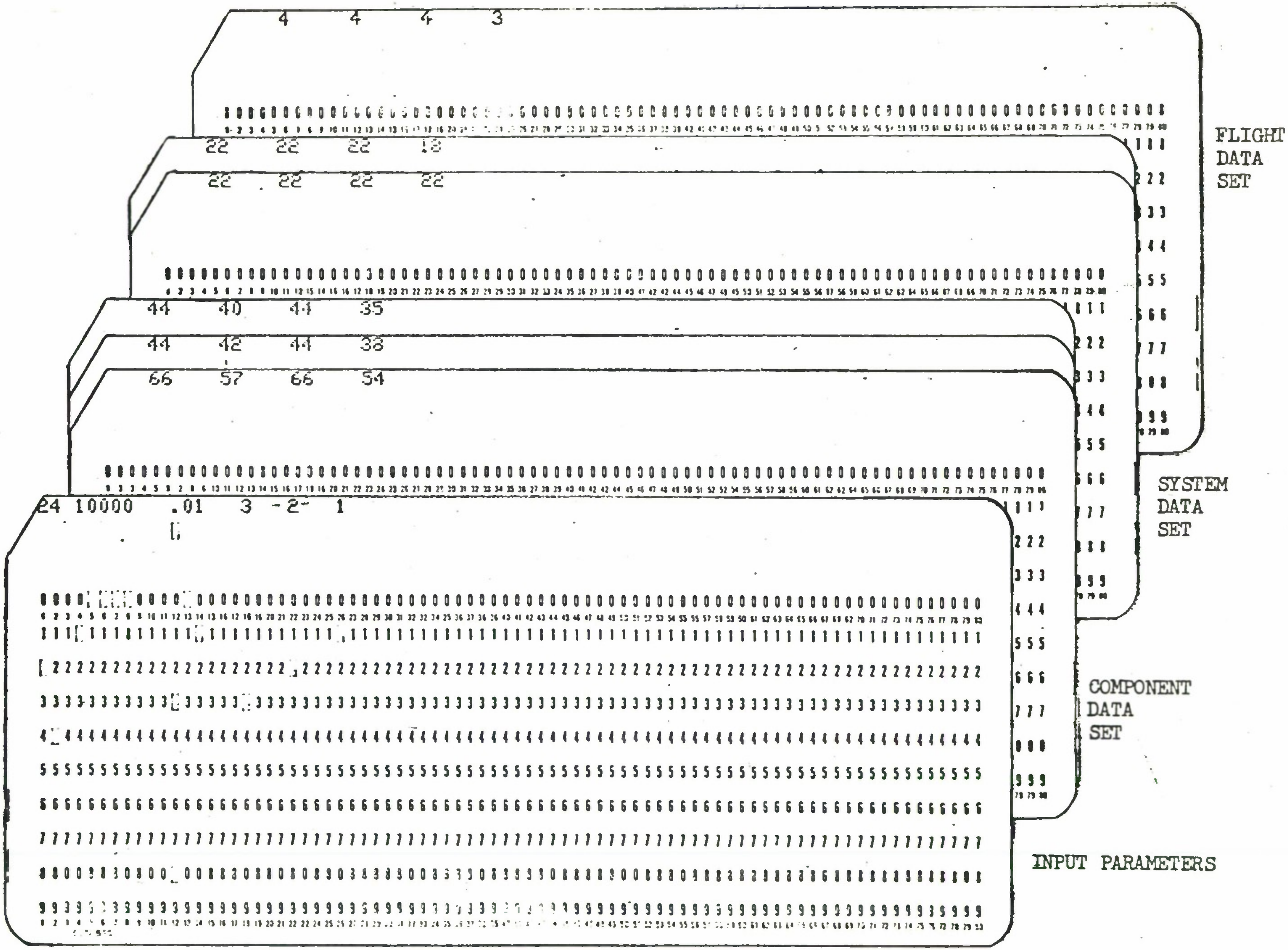
	FIRST ROUND OF TESTING (TIME T_0)		SECOND ROUND OF TESTING (TIME T_1)	
	Number of trials	Number of successes	Number of trials	Number of successes
COMPONENTS:				
TYPE 1	66	57	66	54
TYPE 2	44	42	44	38
TYPE 3	44	40	44	35
LABORATORY SYSTEM DATA:				
FIRST CASE	22	22	22	22
SECOND CASE	22	22	22	18
FIELD SYSTEM DATA (only one case was used for this particular run)	4	4	4	3

Input Data
for
Sample Case

2. Set up a data deck consisting of (refer to fig. 4):

a. One card containing 6 input parameters.

	Columns	Value	
T	1-2	24	It is assumed twenty four months elapsed between the two rounds of testing. (T used only for printout).
NRS	3-8	10000	Number of simulations. Here the circuit equation will be evaluated 10,000 times in determining N_{ceq}, C_{ceq}.
DELTAP	9-14	.01	ΔP- Increment on P This will cause the Beta-distributions and reliability histogram to be formed in steps of .01 along the horizontol-axis, P.
NCOMP	15-18	3	Number of distinct component types. There are three distinct types of components in this circuit even though there is a total of seven components.
NSYDT	19-22	2	Number of cards in the system data set. Since there are two cases of laboratory system data, there must be two data cards in the system data set, as governed by this 2.
NFLDT	23-26	1	Number of cards in flight data set. For this particular run there is only one case of flight data and consequently there is only card in the flight data set.



DATA DECK FOR SAMPLE CASE

FIGURE 4

b. Component Data Set - consists of three data cards, one for each of the three distinct component types in the sample circuit. Card 1 contains the data for the component type labeled #1 in the circuit diagram and circuit equation; card 2 contains the data for component type 2; and card 3 contains data for component type 3. This correspondence between component numbering in the circuit diagram and circuit equation and the ordering of cards in the data deck is essential, however the initial assignment of numbers to the different component types in the circuit is completely arbitrary. Each card in this data set contains the appropriate four values taken from the table of test results above. For example, card 1 contains the number of trials 66 and corresponding number of successes, 57, for component 1 from the first round of testing at time T_0, followed by the test results from round two, time T_1: 66, 54. The numbers are typed in the first 24 columns (6 columns per number, right-adjusted format-4I6).

c. System Data Set - consists of two data cards: one for each of the two cases of laboratory system tests. For example, the first card in this set contains the results of the first case compiled during the first round of tests, time T_0 - 22 trials, 22 successes, followed by those of the second round of tests, time T_1 - 22 trials, 22 successes. The format of the cards in this set is identical to that of the cards in the component data set.

d. Flight Data Set - this data is set up exactly like the previous two sets except that the data are the results of testing the

system in the field.

The output of program RELY as exhibited in the output listing of the sample case, Figure 5, consists of a listing of the input values in tabular form which shows the order in which the system and flight data are evaluated. The system reliability histogram showing the distribution of the R_s values calculated from the circuit equation is exhibited next. The mean and variance are also printed from which N_{ceq} and C_{ceq} are computed. Finally the calculated values for all the A_{ieq} and intermediate variables as well P_{best}, Mu and Variance for all cases of system and flight data combinations are listed. The printout of the histogram begins with the line preceeding the first appearance of a non-zero element in the distribution. For example, in the output of the sample case, the smallest R_s value computed in the 10,000 simulations fell in the range $.77<R_s\leq.78$, so the first line printed was for P = .77.

LAB COMPONENT INPUT DATA   T = T1 - T0 = 24

COMPONENT	T0 N	T0 C	T1 N	T1 C
1	66	57	66	54
2	44	42	44	38
3	44	40	44	35

SYSTEM AND FLIGHT INPUT DATA

		T0				T1			
		SYSTEM		FLIGHT		SYSTEM		FLIGHT	
		N	C	N	C	N	C	N	C
CASE #	1	22	22	4	4	22	22	4	3
CASE #	2	22	22	4	4	22	18	4	3

RELIABILITY HISTOGRAM FOR EQUIVALENT-SYSTEM FOR COMPONENTS

P	DISTRIBUTION
C.77C	0
0.78C	1
C.79C	0
0.80C	0
C.81C	2
0.82C	1
C.83C	4
C.84C	3
0.85C	18
0.86C	34
C.87C	37
C.88C	90
0.89C	158
C.90C	283
C.91C	488
C.92C	760
0.93C	1132
C.94C	1511
C.95C	1774
C.96C	1820
C.97C	1318
C.98C	498
C.99C	67
1.0CC	1

MEAN = 0.93930 D 0C
VARIANCE = C.54983 D-03

FIGURE 5

```
NUMBER OF SIMULATIONS =  1CCCC


CASE #    1  NCEQ = 100.68944  CCEQ =  95.45660  NSEQ =  44.00000  CSEC =  44.00000  NCSEQ =  76 82919  CCSEQ =  75.12305
             NFEQ =   6.36160  CFEQ =   5.36160   NEQ =  17.40282   CEC =  16.15763
            PBEST =   C.92E45    MU =   0.88429   VARIANCE =   0.00502


CASE #    2  NCEQ = 100.68944  CCEQ =  95.45660  NSEC =  25.53925  CSEQ =  21.53925  NCSEQ =  43 85685  CCSEQ =  38.90488
             NFEQ =   6.36160  CFEQ =   5.36160   NEQ =  28.87628   CEC =  25.33410
            PBEST =   C.87733    MU =   0.E5289   VARIANCE =   0.00394
```

FIGURE 5 (Continued)

REFERENCE

1. Naval Ammunition Depot, Oahu, Hawaii, 21 January 1969, "A Proposed Tri-Service Approach for Reliability Assessment", Appendix D.

APPENDIX I

```
      IMPLICIT REAL*8 (A,B,E-H,C-S,U-Z)
      DOUBLE PRECISION N0,N1,C0,C1,KA,KB,KC,KD,KE,K,KSE,KSEPRM,NCEQ,
     1CCEQ,NSEQ,CSEQ,NFEQ,CFEQ,NCSEQ,CCSEQ,NEQ,CEQ,MU,INTERP,
     1XC,YC,X1,Y1,DELTAP
      INTEGER CC0(100),CC1(100),CS0(100),CS1(100),CF0(100),CF1(100),
     1T,DIST
      DIMENSION NC0(100),NC1(100),NS0(100),NS1(100),NF0(100),NF1(100),
     1 DIST(101),BETA(101),A(101),PB(101),AREA(10),1GI),P(101),Y(101)
      EQUIVALENCE (P(1), Y(1))
      COMMON DELTAP /BLKI/ AREA
C
C                      FORMAT STATEMENTS
C
  200 FORMAT (I2,I6,F6.4,3I4/(4I6))
  201 FORMAT(1H1,                                              25H LAB
     1 COMPONENT INPUT DATA,15X,14HT = T1 - T0 = ,I3//64X,2HT0,30X,2HT1/
     1 29X,9HCOMPONENT,19X,1HN,15X,1HC,15X,1HN,15X,1HC)
  202 FORMAT (1H , 33X, I3, 6X, 4(11X, I5))
  203 FORMAT(1H , 58HRELIABILITY HISTOGRAM FOR EQUIVALENT-SYSTEM FOR COM
     1PONENTS// 42X, 1HP, 41X,     12HDISTRIBUTION/(39X,F6.3,41X,I5))
  204 FORMAT (1H ,/  33X, 7HCASE # ,I4, 5X,2(2I6, 2X,2I6, 4X))
  206 FORMAT(1H ,28HSYSTEM AND FLIGHT INPUT DATA/64X,2HT0,28X,2HT1/45X,
     12(10X, 6HSYSTEM, 8X,6HFLIGHT)/ 47X,2(2( 7X,6HN      ,1HC), 2X))
  207 FORMAT (1H , 23HCOMPUTED GC TO OVERFLOW)
  208 FORMAT (1H , 16HDIVISION BY ZERO)
  210 FORMAT (1H ,/ 83X, 7HMEAN = , E12.6/ 79X, 11HVARIANCE = , E12.6)
  212 FORMAT (1H1)
  213 FORMAT(1H ,    7HCASE # ,I4, 2X,7HNCEQ = ,F9.5, 2X,7HCCEQ = ,F9.5,
     1 2X,7HNSEQ = ,F9.5, 2X,7HCSEQ = ,F9.5, 2X,8HNCSEQ = ,F9.5, 2X,
     18HCCSEQ = ,F9.5//14X,7HNFEQ = ,F9.5,2X,7HCFEQ = ,F9.5,3X,6HNEQ = ,
     1 F9.5, 3X,6HCEQ = ,F9.5// 13X,8HFBEST = ,F9.5, 4X, 5HMU = ,F9.5,
     1 3X,11HVARIANCE = ,F9.5)
  214 FORMAT (1H , 24HNUMBER OF SIMULATIONS = , I6)
  250 FORMAT (1H ,/////)
C
C                                INPUT
C
    1 READ (5,200)T,NRS,DELTAP,NCOMP,NSYDT,NFLDT,(NC0(I),CC0(I),NC1(I),
     1CC1(I),I=1,NCOMP),(NS0(I),CS0(I),NS1(I),CS1(I),I=1,NSYDT),(NF0(I),
     1CF0(I),NF1(I),CF1(I),I=1,NFLDT)
C
C T IS TIME(MONTHS) BETWEEN TESTING, NRS IS DESIRED NUMBER OF RS VALUES TO BE
C COMPUTED, DELTAP IS THE DESIRED INCREMENT OF P, NCOMP IS NUMBER OF
C DISTINCT COMPONENT TYPES, NSYDT IS NUMBER OF LAB SYSTEM
C DATA SETS, NFLDT IS NUMBER OF FLIGHT DATA SETS, NC0, CC0, NC1, CC1 IS
C COMPONENT DATA, NS0, CS0, NS1, CS1 IS SYSTEM DATA, NF0, CF0, NF1, CF1
C IS FLIGHT DATA.
C
      WRITE(6,201) T
      DO 109 I=1,NCOMP
  109 WRITE (6,202) I,  NC0(I), CC0(I), NC1(I), CC1(I)
      WRITE (6,250)
      WRITE(6,206)
      NCASE = 0
      DO 2 I=1,NSYDT
      DO 2 J=1,NFLDT
      NCASE = NCASE + 1
    2 WRITE (6,204)NCASE,NS0(I),CS0(I),NF0(J),CF0(J),NS1(I),CS1(I),
```

```
              INF1(J),CF1(J)
            3 DO 104 I=1,NCOMP
              NO = NO0(I)
              CO = CO0(I)
              N1 = NC1(I)
              C1 = CC1(I)
              KA = K(NO,CO,N1,C1)
              SLB1 = NO*KA + N1
              SLB2 = CO*KA + C1
C
C SETUP COMPONENT RELIABILITY HISTOGRAM FOR ITH COMPONENT
C
      CALL BETAF(SLB1,SLB2,BETA,J,PB)
C
C CALL SUBROUTINE TO PRODUCE CUMULATIVE FOR ITH COMPONENT
C
      CALL DCSFIDELTAF,BETA,A,J)
C
C PUT THE VECTOR, A, OF INTEGRAL VALUES PRODUCED IN THE SUBROUTINE FOR THE ITH
C COMPONENT INTO THE ITH ROW OF THE AREA ARRAY.
C
      DO 104 N=1,J
  104 AREA(I,N) = A(N)
C
C INITIALIZE STORAGE AREAS AND VARIABLE LOCATIONS.
C
      DO 105 M = 1,J
  105 DIST(M) = 0
      SMRS = 0.0
      SMRSS2 = 0.0
      NCEQ=0.0
      CCEQ=0.0
      RSMEAN=0.0
      VAR=0.0
      IJ=J
C
C BEGIN MONTE CARLO PROCESS WITH WHICH TO DEVELOP  THE EQUIVALENT-SYSTEM
C RELIABILITY DISTRIBUTION FOR COMPONENTS
C
      DO 111 L=1,NRS
      RS = 0.0
      DO 107 I=1,NCOMP
      YFL = RDM(I)*AREA(I,J)
C
C THE RANDOM NUMBER SELECTED FOR THE ITH COMPONENT IS MULTIPLIED BY THE
C LARGEST VALUE IN THE APPROPRIATE CUMMULATIVE AS A SELF-ADJUSTING
C CALCULATION  TO ACCOUNT FOR DIGITAL ROUNDOFF ERROR
C
  107 P(I) = INTERP(I,J,PB,YFL)
C
C ****************************************************************************
C
C MATH MODEL - BEGIN
      RS = (1.0-(1.0-P(1))**3)*(1.0-(1.0-P(2))**2)*(1.0-(1.0-P(3))**2)
C MATH MODEL - END
C
C ****************************************************************************
C
```

```
      SMRS = SMRS + RS
      SMRSSQ = SMRSSQ +RS**2
      SAVE = L
C
C SET UP RELIABILITY DISTRIBUTION, DIST(M)
C
      M = J + 1
   12 M = M - 1
      IF(1.0-RS .LT. DELTAP) GO TO 108
      IF(PB(M) - RS)13,14,12
   14 IF(RS .GT. 0.0 .OR. RS .EQ. 1.0) GO TO 108
   13 M = M + 1
  108 DIST(M) = DIST(M) + 1
      IF(M .LT. IJ) IJ=M
  111 CONTINUE
      IJ=IJ-1
      WRITE (6,250)
      WRITE (6,203) ((PB(JJ), DIST(JJ)), JJ=IJ,J)
      RSMEAN = SMRS/SAVE
      VAR = (SMRSSQ - 2.0*RSMEAN*SMRS + RSMEAN**2*SAVE)/SAVE
      WRITE (6,210) RSMEAN, VAR
      WRITE (6,250)
      WRITE (6,214) NRS
      IF (VAR)51,51,16
   16 X = RSMEAN
C
C COMPUTE N-EQUIVALENT AND C-EQUIVALENT FOR COMPONENTS
C
      CCEQ = (X**2*(1.0 - X))/VAR - (1.0 + X)
      NCEQ = X*(1.0 - X)/VAR - 3.0
      NCASE = 0
C
C DEVELOP  EQUIVALENT SYSTEM FOR LAB SYSTEM DATA AND COMBINE IT WITH
C EQUIVALENT SYSTEM FOR LAB COMPONENT DATA, YIELDING C,S EQUIVALENT SYSTEM
C
      DO 113 I=1,NSYDT
      NO = NSO(I)
      CO = CSO(I)
      N1 = NS1(I)
      C1 = CS1(I)
      NSEQ=0.0
      CSEQ=0.0
      NCSEQ=0.0
      CCSEQ=0.0
      IF(NO .EQ. 0.0 .AND. N1 .EQ. 0.0) GO TO 114
      KB = K(NO,CO,N1,C1)
      NSEQ = N1 + NO*KB
      CSEQ = C1 + CO*KB
  114 KC = K(NCEQ,CCEQ,NSEQ,CSEQ)
      KSE = 1.0
      KSEPRM = 1.0
      NCSEQ = KSE*NSEQ + KSEPRM*KC*NCEQ
      CCSEQ = KSE*CSEQ + KSEPRM*KC*CCEQ
C
C DEVELOP  EQUIVALENT SYSTEM FOR FLIGHT DATA AND COMBINE IT WITH C,S
C EQUIVALENT SYSTEM, YIELDING FINAL EQUIVALENT SYSTEM REFLECTING ALL SIGNIFICANT
C DATA FROM WHICH THE BEST ESTIMATE OF SYSTEM RELIABILITY, PBEST, IS COMPUTED
C
```

```
      DO 113 M=1,NFLOT
      NC = NFO(M)
      CC = CFO(M)
      N1=NF1(M)
      C1=CF1(M)
      NFEQ=0.0
      CFEQ=0.0
      NEQ=0.0
      CEQ=0.0
      PAEST=0.0
      ML=0.0
      VRNCE=0.0
      IF(NC .EQ. 0.0 .AND. N1 .EQ. 0.0) GO TO 110
      KO = K(NO,CO,N1,C1)
      NFEQ = N1 + KO*NO
      CFEQ = C1 + KO*CO
  110 KE = K(NCSEQ,CCSEQ,NFEQ,CFEQ)
      NEQ = NFEQ + KE*NCSEQ
      CEQ = CFEQ + KE*CCSEQ
      PAEST = CEQ/NEQ
      ML = (CEQ + 1.)/(NEQ + 2.)
      VRN = ((CEQ + 1.)*(NEQ - CEQ + 1.))/((NEQ + 2.)**2*(NEQ + 3.))
      NCASE = NCASE + 1
      WRITE (6,250)
      WRITE(6,213) NCASE,NCEQ,CCEQ,NSEQ,CSEQ,NCSEQ,CCSEQ,NFEQ,CFEQ,NEQ,
     1CEQ,PAEST,ML,VRN
  113 CONTINUE
      GO TO 1
   50 WRITE (6,207)
      GO TO 52
   51 WRITE (6,208)
   52 CALL EXIT
      END
C
C
C        FUNCTION AREAF(B,J)
C
C        PURPOSE
C           TO CALCULATE THE AREA UNDER A CURVE BY SIMPSON'S RULE.
C
C        DESCRIPTION OF ARGUMENTS
C           B - ONE-DIMENSIONAL EQUIDISTANTLY TABULATED FUNCTION THE
C                  AREA UNDER WHICH IS TO BE COMPUTED.
C           J - THE NUMBER OF VALUES IN B.
C
      DOUBLE PRECISION FUNCTION AREAF(B,J)
      DOUBLE PRECISION B(101),  DELTAB, ODD,EVEN
      COMMON DELTAB
      ODD = 0.0
      EVEN = 0.0
      L = J-3
      DO 100 I=2,L,2
      EVEN = EVEN + B(I)
  100 ODD = ODD + B(I+1)
      AREAF= (DELTAB/3.0*(B(1) +4.0*(EVEN + B(J-1)) + 2.0*ODD+B(J)))
      RETURN
      END
```

```
C
C
C         SUBROUTINE BETAF(N,C,BETA,J,PB)
C
C         PURPOSE
C            TO PRODUCE AN EQUIDISTANTLY TABULATED BETA-DISTRIBUTION
C
C         DESCRIPTION OF ARGUMENTS
C              N,C - INPUT PARAMETERS FOR THE BETA-DISTRIBUTION
C                    EQUATIONS.
C           BETA - ONE-DIMENSIONAL BETA-DISTRIBUTION ARRAY PRODUCED IN
C                    THE SUBROUTINE.
C              J - NUMBER OF VALUES IN BETA.
C             PB - ONE-DIMENSIONAL ARRAY OF P-VALUES USED IN GENERATING
C                    THE BETA DISTRIBUTION
C
      SUBROUTINE BETAF(N,C,BETA,J,PB)
      DOUBLE PRECISION N,C,P,PB(101),G,GAMA,BETA(101),X,DELTAP,A,B,D,E,F
      COMMON DELTAP
      P = 0.0
      J = 1
      PB(1) = 0.0
      G = GAMA(N+2.0,C+1.0,N-C+1.0)
      B=DLOG10(.54D+01) - .77D+02
      IF(C .NE. 0.0) GO TO 1
      IF(N .NE. 0.0) GO TO 3
      NN = 4
      BETA(1) = G
      X = G
      GO TO 15
    1 IF(N .NE. C) GO TO 2
      NN = 1
      BETA(1) = 0.0
      X = G
      GO TO 15
    3 NN = 3
      BETA(1) = G
      X = 0.0
      GO TO 15
    2 NN = 2
      BETA(1) = 0.0
      X = 0.0
   15 J = J + 1
      BETA(J)=0.0
      P = P + DELTAP
      PB(J) = P
      IF(1.0-P .LT. DELTAP) GO TO 45
      GO TO (10,20,30,40,50), NN
   10 A=DLOG10(P)
      E=B/A
      IF(C .GE. E) GO TO 15
      BETA(J) = G*P**C
      GO TO 15
   20 A=DLOG10(P)
      D=DLOG10(1.1D+01 - P)
      E=B/A
      F=B/D
      IF(C .GE. E) GO TO 15
```

*

$$P^c = (I \times 10^A)^c \geq 5.4 \times 10^{-77}$$

$$c \log (I \times 10^A) \geq \log(5.4) - 77$$

$$c \geq \frac{\log(5.4) - 77}{\log(P)}$$

CARD NUMBER

```
      IF(N-C .GE. F) GO TO 15
      BETA(J) = G*P**C*(1.0-P)**(N-C)
      GO TO 15
   30 D=DLOG10(.1D+01 - P)
      F=A/D
      IF(N .GE. F) GO TO 15
      BETA(J) = G*(1.0-P)**N
      GO TO 15
   40 BETA(J) = G
      GO TO 15
   45 BETA(J) = X
      DO 60 JJ=1,J
   60 IF(BETA(JJ) .LT. .1D-50) BETA(JJ) = 0.0D 0
      RETURN
   50 WRITE (6,207)
  207 FORMAT (1H , 23HCOMPUTED GO TO OVERFLOW)
   51 RETURN
      END
C                                                                       DQSF  10
C                                                                       DQSF  30
C        SUBROUTINE DQSF                                                DQSF  40
C                                                                       DQSF  50
C        PURPOSE                                                        DQSF  60
C           TO COMPUTE THE VECTOR OF INTEGRAL VALUES FOR A GIVEN        DQSF  70
C           EQUIDISTANT TABLE OF FUNCTION VALUES.                       DQSF  80
C                                                                       DQSF  90
C        USAGE                                                          DQSF 100
C           CALL DQSF (H,Y,Z,NDIM)                                      DQSF 110
C                                                                       DQSF 120
C        DESCRIPTION OF PARAMETERS                                      DQSF 130
C           H      - DOUBLE PRECISION INCREMENT OF ARGUMENT VALUES.     DQSF 140
C           Y      - DOUBLE PRECISION INPUT VECTOR OF FUNCTION VALUES.  DQSF 150
C           Z      - RESULTING DOUBLE PRECISION VECTOR OF INTEGRAL      DQSF 160
C                    VALUES. Z MAY BE IDENTICAL WITH Y.                 DQSF 170
C           NDIM   - THE DIMENSION OF VECTORS Y AND Z.                  DQSF 180
C                                                                       DQSF 190
C        REMARKS                                                        DQSF 200
C           NO ACTION IN CASE NDIM LESS THAN 3.                         DQSF 210
C                                                                       DQSF 220
C        SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                  DQSF 230
C           NONE                                                        DQSF 240
C                                                                       DQSF 250
C        METHOD                                                         DQSF 260
C           BEGINNING WITH Z(1)=0, EVALUATION OF VECTOR Z IS DONE BY    DQSF 270
C           MEANS OF SIMPSONS RULE TOGETHER WITH NEWTONS 3/8 RULE OR A  DQSF 280
C           COMBINATION OF THESE TWO RULES. TRUNCATION ERROR IS OF      DQSF 290
C           ORDER H**5 (I.E. FOURTH ORDER METHOD). ONLY IN CASE NDIM=3  DQSF 300
C           TRUNCATION ERROR OF Z(2) IS OF ORDER H**4.                  DQSF 310
C           FOR REFERENCE, SEE                                          DQSF 320
C           (1) F.B.HILDEBRAND, INTRODUCTION TO NUMERICAL ANALYSIS,     DQSF 330
C               MCGRAW-HILL, NEW YORK/TORONTO/LONDON, 1956, PP.71-76.   DQSF 340
C           (2) R.ZURMUEHL, PRAKTISCHE MATHEMATIK FUER INGENIEURE UND   DQSF 350
C               PHYSIKER, SPRINGER, BERLIN/GOETTINGEN/HEIDELBERG, 1963, DQSF 360
C               PP.214-221.                                             DQSF 370
C                                                                       DQSF 380
C                                                                       DQSF 400
```

```
      SUBROUTINE DQSF(H,Y,Z,NDIM)                                       DQSF 410
      DIMENSION Y(1),Z(1)                                               DQSF 440
      DOUBLE PRECISION Y,Z,H,HT,SUM1,SUM2,AUX,AUX1,AUX2                 DQSF 450
C                                                                       DQSF 460
      HT=.33333333333333333D0*H                                         DQSF 470
      IF(NDIM-5)7,8,1                                                   DQSF 480
C                                                                       DQSF 490
C     NDIM IS GREATER THAN 5. PREPARATIONS OF INTEGRATION LOOP          DQSF 500
    1 SUM1=Y(2)+Y(2)                                                    DQSF 510
      SUM1=SUM1+SUM1                                                    DQSF 520
      SUM1=HT*(Y(1)+SUM1+Y(3))                                          DQSF 530
      AUX1=Y(4)+Y(4)                                                    DQSF 540
      AUX1=AUX1+AUX1                                                    DQSF 550
      AUX1=SUM1+HT*(Y(3)+AUX1+Y(5))                                     DQSF 560
      AUX2=HT*(Y(1)+3.875D0*(Y(2)+Y(5))+2.625D0*(Y(3)+Y(4))+Y(6))       DQSF 570
      SUM2=Y(5)+Y(5)                                                    DQSF 580
      SUM2=SUM2+SUM2                                                    DQSF 590
      SUM2=AUX2-HT*(Y(4)+SUM2+Y(6))                                     DQSF 600
      Z(1)=0.D0                                                         DQSF 610
      AUX=Y(3)+Y(3)                                                     DQSF 620
      AUX=AUX+AUX                                                       DQSF 630
      Z(2)=SUM2-HT*(Y(2)+AUX+Y(4))                                      DQSF 640
      Z(3)=SUM1                                                         DQSF 650
      Z(4)=SUM2                                                         DQSF 660
      IF(NDIM-6)5,5,2                                                   DQSF 670
C                                                                       DQSF 680
C     INTEGRATION LOOP                                                  DQSF 690
    2 DO 4 I=7,NDIM,2                                                   DQSF 700
      SUM1=AUX1                                                         DQSF 710
      SUM2=AUX2                                                         DQSF 720
      AUX1=Y(I-1)+Y(I-1)                                                DQSF 730
      AUX1=AUX1+AUX1                                                    DQSF 740
      AUX1=SUM1+HT*(Y(I-2)+AUX1+Y(I))                                   DQSF 750
      Z(I-2)=SUM1                                                       DQSF 760
      IF(I-NDIM)3,6,6                                                   DQSF 770
    3 AUX2=Y(I)+Y(I)                                                    DQSF 780
      AUX2=AUX2+AUX2                                                    DQSF 790
      AUX2=SUM2+HT*(Y(I-1)+AUX2+Y(I+1))                                 DQSF 800
    4 Z(I-1)=SUM2                                                       DQSF 810
    5 Z(NDIM-1)=AUX1                                                    DQSF 820
      Z(NDIM)=AUX2                                                      DQSF 830
      RETURN                                                            DQSF 840
    6 Z(NDIM-1)=SUM2                                                    DQSF 850
      Z(NDIM)=AUX1                                                      DQSF 860
      RETURN                                                            DQSF 870
C     END OF INTEGRATION LOOP                                           DQSF 880
C                                                                       DQSF 890
    7 IF(NDIM-3)12,11,8                                                 DQSF 900
C                                                                       DQSF 910
C     NDIM IS EQUAL TO 4 OR 5                                           DQSF 920
    8 SUM2=1.125D0*HT*(Y(1)+Y(2)+Y(2)+Y(2)+Y(3)+Y(3)+Y(3)+Y(4))         DQSF 930
      SUM1=Y(2)+Y(2)                                                    DQSF 940
      SUM1=SUM1+SUM1                                                    DQSF 950
      SUM1=HT*(Y(1)+SUM1+Y(3))                                          DQSF 960
      Z(1)=0.D0                                                         DQSF 970
      AUX1=Y(3)+Y(3)                                                    DQSF 980
      AUX1=AUX1+AUX1                                                    DQSF 990
      Z(2)=SUM2-HT*(Y(2)+AUX1+Y(4))                                     DQSF1000
```

```
      IF(NDIM-5)10,9,9                                                  DQSF1010
    9 AUX1=Y(4)+Y(4)                                                    DQSF1020
      AUX1=AUX1+AUX1                                                    DQSF1030
      Z(5)=SUM1+HT*(Y(3)+AUX1+Y(5))                                     DQSF1040
   10 Z(3)=SUM1                                                         DQSF1050
      Z(4)=SUM2                                                         DQSF1060
      RETURN                                                            DQSF1070
C                                                                       DQSF1080
C     NDIM IS EQUAL TO 3                                                DQSF1090
   11 SUM1=HT*(1.25D0*Y(1)+Y(2)+Y(2)-.25D0*Y(3))                        DQSF1100
      SUM2=Y(2)+Y(2)                                                    DQSF1110
      SUM2=SUM2+SUM2                                                    DQSF1120
      Z(3)=HT*(Y(1)+SUM2+Y(3))                                          DQSF1130
      Z(1)=0.D0                                                         DQSF1140
      Z(2)=SUM1                                                         DQSF1150
   12 RETURN                                                            DQSF1160
      END                                                               DQSF1170
C
C
C        FUNCTION GAMA(ARG1,ARG2,ARG3)
C
C        PURPOSE
C           TO CALCULATE CONSTANT PART OF BETA EQUATION FOR SUBROUTINE
C           BETAF
C
C        DESCRIPTION OF ARGUMENTS
C           ARG1 - CORRESPONDS TO N+2 IN BETA EQUATION
C           ARG2 - CORRESPONDS TO C+1 IN BETA EQUATION
C           ARG3 - CORRESPONDS TO N-C+1 IN BETA EQUATION
C
      DOUBLE PRECISION FUNCTION GAMA (ARG1,ARG2,ARG3)
      DOUBLE PRECISION COEFF,X,Y,Z,ARG1,ARG2,ARG3
      COEFF = 1.0
   11 X = 1.0
      Y = 1.0
      Z = 1.0
      CHECK = 0.0
      IF(ARG1 .LE. 57.0) GO TO 12
      CHECK = 1.0
      ARG1 = ARG1 - 1.0
      X = ARG1
   12 IF(ARG2 .LE. 57.0) GO TO 13
      CHECK = 1.0
      ARG2 = ARG2 - 1.0
      Y = ARG2
   13 IF(ARG3 .LE. 57.0) GO TO 14
      CHECK = 1.0
      ARG3 = ARG3 - 1.0
      Z = ARG3
   14 COEFF = COEFF*(X/Y)*(1.0/Z)
      IF(CHECK .EQ. 1.0) GO TO 11
      CALL GAMMA(ARG1, GX, IEX)
      CALL GAMMA(ARG2, GY, IEY)
      CALL GAMMA(ARG3, GZ, IEZ)
      GAMA = COEFF*(GX/GY)*(1.0/GZ)
      RETURN
      END
C
```

CARD NUMBER

```
C
C        SUBROUTINE GAMMA(XX,GX,IER)
C
C        PURPOSE
C           TO COMPUTE THE GAMMA FUNCTION FOR A GIVEN ARGUMENT
C
C        DESCRIPTION OF ARGUMENTS
C
C           XX - THE ARGUMENT FOR THE GAMMA FUNCTION
C           GX - THE RESULTANT GAMMA FUNCTION VALUE
C          IER - RESULTANT ERROR CODE WHERE
C                IER=0 NO ERROR
C                IER=1 XX IS WITHIN .000001 OF BEING A NEGATIVE INTEGER
C                IER=2 XX IS GT 57, OVERFLOW, GX IS SET TO 1.0E75
C
      SUBROUTINE GAMMA(XX,GX,IER)
      DOUBLE PRECISION XX,X,GX,Y,GY
      IF(XX-57.)6,6,4
    4 IER=2
      GX=1.E75
      RETURN
    6 X=XX
      ERR = 1.0D-6
      IER=0
      GX=1.0
      IF(X-2.0)50,50,15
   10 IF(X-2.0)110,110,15
   15 X=X-1.0
      GX=GX*X
      GO TO 10
   50 IF(X-1.0)60,120,110
C
C        SEE IF X IS NEAR NEGATIVE INTEGER OR ZERO
C
   60 IF(X-ERR)62,62,80
   62 B = X
      Y = FLOAT(INT(B)) - X
      IF(DABS(Y) - ERR) 130,130,64
   64 IF(1.0-Y-ERR)130,130,70
C
C        X NOT NEAR NEGATIVE INTEGER OR ZERO
C
   70 IF(X-1.0)80,80,110
   80 GX=GX/X
      X=X+1.0
      GO TO 70
  110 Y=X-1.0
      GY=1.0+Y*(-0.5771017+Y*(+0.9858540+Y*(-0.8764218+Y*(+0.8328212+
     1Y*(-0.5684729+Y*(+0.2548205+Y*(-0.05149930)))))))
      GX=GX*GY
  120 RETURN
  130 IER=1
      RETURN
      END
```

CARD NUMBER

```
C
C
C        FUNCTION INTERP(I,J,F,YFL)
C
C        PURPOSE
C           TO PERFORM A LINEAR INTERPOLATION
C
C        DESCRIPTION OF ARGUMENTS
C              I - COMPONENT TYPE DESIGNATION
C              J - NUMBER OF VALUES IN ARRAY IN WHICH THE NUMBER YFL IS
C                  TO BE INTERPOLATED
C              F - ONE-DIMENSIONAL ARRAY FROM WHOSE ELEMENTS THE INTERPOLATED
C                  VALUE IS TO BE CALCULATED
C            YFL - NUMBER WHOSE CORRESPONDING VALUE IS TO BE FOUND BY
C                  INTERPOLATION.
C
      DOUBLE PRECISION FUNCTION INTERP(I,J,F,YFL)
      DOUBLE PRECISION C,D,X,Y,A,F,YFL,AX(100,101)
      DIMENSION A(101), F(101)
      COMMON /BLK1/ AX
      DO 100 M=1,J
  100 A(M) = AX(I,M)
      L = 0
   10 L = L + 1
      IF(L .GT. J) GO TO 14
      IF(A(L)-YFL)10,11,12
   11 INTERP = F(L)
      GO TO 13
   12 C = A(L) - A(L-1)
      Y = YFL - A(L-1)
      D = F(L) - F(L-1)
      X = Y/C*D
      INTERP = F(L-1) + X
   13 RETURN
   14 WRITE (6,200)
  200 FORMAT(1H ,'OVERFLOW - INTERP')
      INTERP=F(J)
      RETURN
      END
C
C
C        FUNCTION K(N0,C0,N1,C1)
C
C        PURPOSE
C           TO PRODUCE A WEIGHTING FACTOR FOR DEGRADING THE LESS
C           SIGNIFICANT DATA N0, C0.
C
C        DESCRIPTION OF ARGUMENTS
C           N0,C0 - NUMBER OF TRIALS AND SUCCESSES (OR ANALOGOUS
C                   PARAMETERS)  -  LESS SIGNIFICANT DATA
C           N1,C1 - NUMBER OF TRIALS AND SUCCESSES (OR ANALOGOUS
C                   PARAMETERS)  -  MORE SIGNIFICANT DATA
C
      DOUBLE PRECISION FUNCTION K(N0,C0,N1,C1)
      DOUBLE PRECISION N0,C0,N1,C1,N,C,BTX(101),BTZ(101),BX(101),CMAREA,
     1TIAREA,AREA,DELTAP,CX(101),FR(101)
      COMMON DELTAP
      K = 1.0D 0
```

```
      IF(NC .EQ. N1 .AND. C0 .EQ. C1) GO TO 2
      IF(NC .EQ. 0.0 .OR. N1 .EQ. 0.0) GO TO 2
      CALL BETAF(NC,C0,BTX,J,PP)
      CALL BETAF(N1,C1,BTZ,J,PP)
      DO 100 L=1,J
      IF(BTZ(L) .LE. BTX(L)) GO TO 1
      BX(L) = BTX(L)
      GO TO 100
    1 BX(L) = BTZ(L)
  100 CX(L) = BTZ(L)
      CMAREA = AREAF(BX,J)
      TIAREA = AREAF(CX,J)
      K = CMAREA/TIAREA
    2 RETURN
      END
```

**** E N D   O F   D A T A ****

/*

TABLE OF DISTRIBUTION

	COPY NO.
COMMANDER DEFENSE DOCUMENTATION CENTER ATTN: TIPCR CAMERON STATION ALEXANDRIA, VIRGINIA 22314	1-20
COMMANDING GENERAL U. S. ARMY MATERIEL COMMAND ATTN: AMC-QA WASHINGTON, D. C. 20315	21
COMMANDING GENERAL U. S. ARMY MATERIEL COMMAND ATTN: AMCRD-R WASHINGTON, D. C. 20315	22
COMMANDING GENERAL U. S. ARMY MUNITIONS COMMAND ATTN: AMSMU-QN AMSMU-RE-R AMSMU-M DOVER, NEW JERSEY 07801	23-25
COMMANDING OFFICER U. S. ARMY EDGEWOOD ARSENAL ATTN: SMUEA-QAER EDGEWOOD ARSENAL, MARYLAND 21010	26
COMMANDING OFFICER U. S. ARMY FRANKFORT ARSENAL ATTN: SMUFA-Q2100 PHILADELPHIA, PA. 19137	27
COMMANDING OFFICER FORT DETRICK ATTN: SMUFD-QA FREDERICK, MARYLAND 21701	28
COMMANDING OFFICER U. S. ARMY AMMUNITION PROCUREMENT AND SUPPLY AGENCY ATTN: SMUAP-R JOLIET, ILLINOIS 60436	29

COMMANDING OFFICER 30
QUALITY EVALUATION LABORATORY
U. S. NAVAL AMMUNITION DEPOT
OAHU, HAWAII
ATTN: MR. PETER COLE
C/O NAVY FLEET POST OFFICE
SAN FRANCISCO, CALIFORNIA 96612

DIRECTORATE OF SPECIAL WEAPONS 31
SAN ANTONIO AIR MATERIAL AREA
ATTN: MR. ARMANDO HINOJOSA
KELLY AIR FORCE BASE, TEXAS 78241

COMMANDING OFFICER 32
U. S. ARMY PICATINNY ARSENAL
ATTN: MR. H. W. PAINTER

DATA PROCESSING SYSTEMS OFFICE
ATTN: MR. DAVID L. GROBSTEIN, CHIEF 33

OPERATIONS DIVISION
ATTN: DR. LEONARD F. NICHOLS, CHIEF 34

SOFTWARE CONCEPTS AND SUPPORT BRANCH
ATTN: MR. INGO E. RUCKER, CHIEF 35

MR. EDWARD D. GABEL 36

MR. FORREST L. MCMAINS 37

MISS VIVIAN J. SHEVLIN 38

SCIENTIFIC AND ENGINEERING APPLICATIONS DIVISION
ATTN: MR. ROBERT I. ISAKOWER, CHIEF 39

MR. WALTER G. MARPLE 40

MR. JOSEPH J. SIERODZINSKI 41

MATHEMATICAL ANALYSIS DIVISION
ATTN: MR. BRUCE D. BARNETT, CHIEF 42-45

MR. JAMES G. BEVELOCK 46

MR. GEORGE A. BRINCKA 47

MRS. MIRIAM HARRIS 48

ARTILLERY AMMUNITION LABORATORY
SPECIAL AMMUNITION AND ANALYSIS SECTION
EVALUATION UNIT
ATTN: MR. ANTHONY J. RICCIARDI 84

TECHNICAL INFORMATION BRANCH
ATTN: MR. JOHN P. NOONAN 85-89

UNCLASSIFIED
Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author)

Picatinny Arsenal
Dover, New Jersey

2a. REPORT SECURITY CLASSIFICATION

UNCLASSIFIED

2b. GROUP

3. REPORT TITLE

RELY - A program for estimating overall system reliability based on component system and flight data

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

5. AUTHOR(S) (First name, middle initial, last name)

Thomas A. Neef

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
March 1969	48	1

8a. CONTRACT OR GRANT NO.	9a. ORIGINATOR'S REPORT NUMBER(S)
b. PROJECT NO.	Technical Memorandum 1891
c.	9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)
d.	

10. DISTRIBUTION STATEMENT

Statement 1: Distribution of this document is unlimited.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY

13. ABSTRACT

This report documents a program for estimating overall system reliability by judiciously combining individual component data, laboratory test data and finally flight data as gathered at two different periods in time. All data used in arriving at this single reliability estimate is assumed to be attribute in nature.

DD FORM 1 NOV 65 1473 REPLACES DD FORM 1473, 1 JAN 64, WHICH IS OBSOLETE FOR ARMY USE.

UNCLASSIFIED
Security Classification

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
RELY Reliability Attribute data						